

AF 119 27W

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Karen R. Klutz
Sandeep K. Singhal
Thyra Rauch
David A. Schell

GAU: 2179

Examiner: **Truc T. Chuong**

Confirmation No.: **2541**

Customer No.: **23307**

Attorney Docket No.: **RSW-00-0021**

Application No.: **09/614,852**

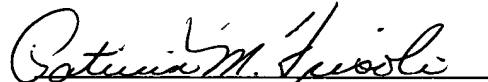
Filed: **July 12, 2000**

For: **Method and Apparatus for
Determining Computer Interface
Display Attributes**

CERTIFICATE OF MAILING

I hereby certify that this correspondence, along with any paper indicated as being enclosed, are being deposited with the United States Postal Service as first-class mail, postage pre-paid, on August 12, 2005 in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

August 12, 2005
Date


Patricia M. Frisoli

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Attention: Board of Patent Appeals and Interferences

**RESPONSE TO THE EXAMINER'S ANSWER OF JUNE 16, 2005
TO APPEAL BRIEF FILED ON MARCH 30, 2005**

This brief is in response to the Examiner's Answer dated June 16, 2005 to Appellant's Appeal Brief filed on March 30, 2005.

I. Grouping of Claims

As a preliminary matter, the Examiner, citing 37 C.F.R. 1.192(c)(7), asserted that all of claims 1-29 stand or fall together because appellant's brief does not include a statement that this grouping of claims does not stand or fall together.

This is erroneous. The Examiner is applying old rules. Under the new appeal rules, particularly 37 C.F.R. 41.37, there no longer is any requirement to separately group the claims. Instead, in the Argument section, "For each ground of rejection applying to two or more claims, the claims may be argued separately or as a group". 37 C.F.R. 41.37(c)(viii)

Appellant has, in fact, presented arguments of separate patentability with respect to dependent claims 12 and 25, from which claims 13-20 and 26-29 depend. Accordingly, claims 12-20 and 25-29 must be separately considered from claims 1-11 and 21-24.

II. Reply to Substantive Arguments

The argument set forth under section (10) of the Examiner's answer is a verbatim copy of the rejections set forth in the final office action. Accordingly, they have already been addressed and require no further comment.

In section (11) entitled ***Response to Argument***, however, the Examiner has presented new responses.

In this section, the Examiner allegedly addresses three separate arguments made by appellant. However, it will be shown below that (a) appellant did not make any of those three arguments, (b) the alleged arguments are not even pertinent to the present invention as claimed, and (c) the Examiner's responses to the alleged arguments are not relevant to the present invention as claimed.

A. Appellant's First Alleged Argument

First, the Examiner states that "appellant has argued that Lucus fails to teach or suggest the first document and the other documents having a same file type". It is

unclear to which of appellant's arguments the Examiner is referring. There does not appear to be any argument in appellant's appeal brief remotely resembling this alleged argument. Certainly, appellant does not deny that two documents can have the same file type. The closest argument to this statement that appellant made is that Lucas does not teach applying the display attributes of a first document to a second document having the same file type.

In any event, in responding to this alleged argument, the Examiner specifically argued:

Lucas allows a grouping of the documents so that they can be manipulated as groups, and the user can specify a grouping of documents by using a find tool 100 to send requests to receive the documents (col. 14 lines 51-60, and col. 18 lines 45-57). The system compares if the stored and requested attributes are matched. If they are matched, all documents or files with the same extension or file type will be grouped together into a specific display shape (col. Nine lines 15-45, col. 10 lines 54-55, col. 18 lines 45-50, and figs. 1, 3, 5 & 9. Therefore, all files with the given extension or common file type will be grouping together.

The portions of the specification that the Examiner refers to concerns the "strand" feature in Lucas.

Strands are a system for positioning screen objects in a three-dimensional workspace. Strands allow grouping of documents, so that they can be manipulated as groups. Strands are a method of applying constraints to the organization of screen objects in three dimensions. Lucas, column 9, lines 15-19. In one example, the documents comprising a strand can be collected using a Find tool. For instance, the Find tool may be used to locate documents containing a particular string of characters. When the Find tool is used, the documents found to contain the string are displayed along the strand 15. The Find tool is the parent of that strand. Lucas, column 10, line

66-column 11, line 7.

In accordance with this feature, files can be grouped together. In fact, they can be grouped together by file type. Furthermore, they can be displayed collectively with a single command. However, this has nothing to do with applying the display attributes of one file to another file having the same file type. In Lucas' strand feature, a group of documents are defined as a "strand" and can then be manipulated collectively. There is nothing in the disclosure of Lucas that suggests anything but the fact that each document in the strand is opened and displayed using its own attributes.

Accordingly, the Examiner's "response" is irrelevant in that it responds to a nonexistent, irrelevant argument allegedly made by appellant and addresses an issue that is irrelevant to the present claims.

B. Appellant's Second Alleged Argument

Second, the Examiner asserted that appellant had argued that "Lucas fails to further teach or suggest when the file type of the first document is opened, accessing the stored data indicating the values of other documents having the same file type as the first document".

Once again, appellant made no such argument and it is unclear to which of appellant's actual arguments the Examiner is referring. Appellant's best guess is that the Examiner is referring to appellant's argument pertaining to the dependent claims 12 and 25 concerning the supplemental feature of opening a second file of a certain file type whenever a first file of another file type is opened, the second file having the same defined relationship with the first file as two files previously displayed simultaneously.

In response to this alleged argument, the Examiner argues:

Lucus teaches that the documents in the system consist of attributes, each attribute having a name and a value, which are used to define the file type of each document in order to determine what operations are permissible, or what kinds of applications can be used for with [sic] that type of grouping documents (col. 5 lines 10-17). Lucas also teaches the documents are stored in repositories (a remote database or other servers throughout the Network), the repositories respond with one or more messages containing the unique identifier(s) or UID (which is necessary and sufficient to refer to a specific document) of documents that match the description in the search request from the user to create a display of the document on the display device (Summary, col. 1 line 64-col. 2 line 32).

Again, the Examiner is referring to the "strand" feature of Lucas. Once again, the Examiner is missing the point. The language of claims 12 and 25 does not read on a group of documents that have been grouped together by a search criterion into a specific display shape. Specifically, using claim 12 as an example, it recites "(4) displaying a second displayable file simultaneously with said first file on said computer display in a manner selected by said operator; (5) storing data associated with said type of said first file indicating at least a type of said second file relative to said first file; and (6) when another file of the type of said first file is opened for display, automatically opening another file of the same type as said second file having the same relationship to said another file of said first type as said second file had to said first file".

Lucas' "strand" feature is nothing like this. In Lucas, multiple files are opened simultaneously in particular positions on the screen because they have been grouped together previously using the "strand" feature, not because the system has stored "data associated with said type of said first file indicating at least a type of said second file relative to said first file; and... when another file of the type of said first file is opened for display, automatically opening another file of the same type as said second file having the same relationship to said another file of said first type as said second file had to

said first file". There is no data stored in Lucus in connection with the strand feature indicating a type of a second file relative to the type of the first file nor is there any step of automatically opening another file because it has the same relationship. The files in a Lucus strand are opened simultaneously and in particular positions on the display screen because the user has previously grouped them together as a strand and told the system how and where to open the files in the strand.

C. Appellant's Third Alleged Argument

Third, the Examiner asserted that appellant had argued that "Lucus fails to teach opening a group of files with one command".

Once again, appellant is at a loss as to which of appellant's actual arguments the Examiner is referring.

In responding to this supposed argument, the Examiner stated:

Lucus clearly shows the user can specify a grouping of documents by using the find tool 100 to send requests to retrieve the wanted documents (col. 14 lines 51-60, and col. 18 lines 45-57). The system compares whether the stored in requested attributes are matched. If they are matched, all documents or files with same extension or file type will be grouped together into a specific display shape (col. 9 lines 15-45, col. 10 lines 54-55, col. 18 lines 45-50, and figs 1, 3, 5 & 9).

Appellant has not argued that Lucus fails to teach opening a group of files with one command. Lucus clearly teaches this. However, its relevance to the claims of the present application is not known. The fact that Lucus can open a group of files with one command does not read on the language of claim 1, which does not simply recite opening multiple files with one command. Rather, it specifically recites "(3) when another file of the type of said first file is opened by an operator for display, accessing

said stored data indicating should value of said at least one attribute; and (4) displaying said another file of the type of said first file using said stored value of said at least one attribute". This is quite a bit more specific of a recitation.

III. Conclusion

The Examiner has presented no new argument that would compel upholding the present claim rejections.

Respectfully submitted,

Dated: August 12, 2005



Theodore Naccarella
Registration No. 33,023

Synnestvedt & Lechner
2600 Aramark Tower
1101 Market Street
Philadelphia, PA 19107
Telephone: (215) 923-4466
Facsimile: (215) 923-2189

TXN:pmf